Where are these amateur game programmers?

Danny van der Kleij

Piet Zwart Institute
Networked Media Program

Users and Abusers
10 December 2010

1. Introduction

For the 2004 edition of Sherry Turkle's The Second Self, 20 years after it's initial release, Sherry Turkle wrote a foreword about the difference's in computer culture between 1984 and 2004.[1] In this foreword she states:

> [...] Yet there are some things that have not been carried forward[...] So, too, is a vibrant culture of personal computer owners who built and bought home computers for the joy of understanding how they worked. But in today's cultural mainstream, these actors are no longer with us. The socially shared activity of computer programming and hardware tinkering has been displaced by playing games, participation in online chat and blogs, and using applications software out of the box. (Turkle, 2004, p 7)

Although this quote about the disappearance of certain aspects of computer culture spans more than just programming, it made me wonder about amateur game programming. In the 1980's hardware tinkering and computer programming was closely related to amateur game development. Has this culture of amateur game design disappeared as well? If it still exists, is it also a socially shared activity? The first thing that comes to mind, is the rising level of complexity that game development has had and still is increasing at the moment. Amateur game developers simply being unable to keep up with industry standards seems a simple and straightforward explanation for this disappearance. Has programming become less of a practice within the amateur game world or has it merely moved toward other sections of game development? We also see amateurs do 3D modelling and texture design or altering the rules, or designing new features for an already existing game? Users of games have always been prone to personalize their gaming experience. Since the mid 90's gamers have taken an active role in altering their own game-play not only by voicing their critique to commercial game development companies but also by producing their own game modifications.[2]

When thinking of a game developed by amateurs, we tend to think of a game that is old fashioned, small and simple. With the recent rise of mobile phones as gaming platforms, as well as the advertisement ridden world of online flash games, we see a lot of these games. However these simple games are not only produced by amateurs. More often they are made by small companies. From playing a flash game it can be hard to see whether it is made by an amateur or one of these small companies. If an amateur flash game developer added an advertisement to his game and thus generate a small form of income, does this make him a professional?

[1] The Second Self (p. 8)
[2] WADs, Bots and Mods: Multiplayer FPS Games as Co-creative Media (p. 1)

In defining the amateur, I don't think it is fair to solely base the definition on whether profit is made or a certain rate of success is gained. Being one of the first to release a game for a new platform, for example a new mobile phone is not something, in my mind, an amateur would do. For me the amateur is someone who does his practice because he can't not do it. In the column 'The Amateur': Amateur vs Indie on gamesetwatch.com professional game developer Andrew Doull[3] sums it up quite nicely:

> [...]subversive, modded, copycat, patched together from pre-built parts, non-commercial or anti-commercial. Amateur game development is done by people who are scratching an itch, who can't not write computer games, who want to see their ideas in pixel form ahead of trying to generate a return. It might be because their favourite game or game genre has been abandoned as a commercial enterprise. Or because they have an idea so out there, so unachievable, so unplayable that no attempt to commercialise the game could possibly succeed. (Doull, 2008, p2)

Reading Doull's statements on amateur game development, I think that the amateur's goals are very personal. For example when the amateur will release his game to the outside world, mostly to a small community of individuals, longing for or even creating the same kind of game. The amateur releases his game whenever the amateur himself feels like the game is finished.

2. Modification as a personal creation

The type of amateur game design we see in mainstream media is the culture of modification or mod making. The reason behind this can be sought in the fact that commercial game development has an interest in what kind of game-play the users like, in order to release more popular games in the future.
Commercial Game developers actively support this by releasing open source software with which one can modify the game, its levels, its options or its rules. Some game developers even go as far as to release the source code of the game. Mostly this release is done via the website of the game developers themselves.
In 1993 Id Software released Doom. In contrary to their previous release Wolfenstein 3d (1992), Id software developers had opted to separate game media files(mostly sound and texture files) from the games source code. These media files could then easily be accessed via a folder where the game was installed on the hard disk drive. This made modification of the media files easier and the original source code of the game did not require modification.[4]
Coming back to Turkle's statement on the disappearance of computer programming as a socially shared activity, I think the move of Id Software to give easy access to game media fit right into the context of this statement. However Id Software also released the source code to Doom, in order for its users to work on a complete overhaul of the game.[5]

[3] http://www.gamesetwatch.com/2008/02/column_the_amateur_amateur_vs.php
[4] Creative User-centred Design Practices: Lessons from Game Cultures(p. 109)
[5] WADs, Bots and Mods: Multiplayer FPS Games as Co-creative Media (p. 2)

One of the most game changing but also one of the most successful game modifications is DotA, shorthand for Defence of the Ancients. A mod for Blizzard's significantly less successful Warcraft 3.[6]
Even though DotA has now multiple commercial spin offs, it's initial success has come from the fact that the development of updates went parallel with releasing new content. The regularity of updates not only kept the user base interested, it also opened a way for the users of DotA to demand certain changes. On May 30th of 2008 Mike Walbridge wrote on gamesetwatch.com: "Over at the "official" *DOTA Allstars* forums as I write this, there are 800 people logged in and over 100,000 total topics and over 23,000 topics in the general forum in the last month"
This never ending stream of extra content and game community driven refining of game-play leads us back to Sherry Turkle and the promise of computer games "The first is a touch of infinity—the promise of a game that never stops."[7]
Another almost completely community driven game is the 2 dimensional fighting game M.u.g.e.n.(or Mugen) by indie developer Elecbyte. The main idea behind Mugen is that:

> [...]The engine allows users to insert created characters, background stages, and other game objects through interpreted text files, graphics, and sound compilations to create a functioning fighting game similar to commercial games.[8]

The result is almost a collage of fighting game culture, user made imports of other fighting games are mixed with characters from pop culture as well as the user's own creations.

Streetfigter's Vega fighting King Godzilla in M.u.g.e.n. Gamesradar.com (June 2009)

[6] http://www.gamesetwatch.com/2008/05/column_the_game_anthropologist_defense_of.php
[7] The New Media Reader(p. 511)
[8] http://en.wikipedia.org/wiki/M.U.G.E.N

As Id Software did with Doom, Elecbyte released all sorts of tools (mainly Dos command-line tools) as well as tutorials and documentation on how to modify and create the game. One of the most significant differences with the modification culture behind Doom is that without user created content the game Mugen, hardly exists at all. User generated content is the core of the Mugen game, and its game-play. Translated from Japanese the word mugen means endless, dreams, visions or fantasy. It is the idea of endless creativity, the promise of the ultimate game that sets Mugen apart from other fighting games. Testing out the most recent changers, on your own or others with is part of the standard game experience.
I think that by motivating the users of Mugen to create the game itself in almost every aspect of the game, Elecbyte created a system that eventually collapsed upon itself. By it's very nature Mugen encouraged modification, and for most users this means an improvement to their own game. The only thing Elecbyte did not do, was release the source code of the game.[9] When Elecbyte stopped the support and innovation of their game in 2003, the amateur community around Mugen simply carried on without them. Not only was a beta version of Mugen hacked so that it worked in full, the amateur programming community lacking the original source code also created at least 7 clone projects, in order to go on without Elecbyte. These clone projects either had full support for the content the Mugen community already made or had the same kind of techniques to create new features for the game or both. Following my definition of amateur these clone projects are full standalone games programmed by a community of amateurs. Whether Elecbyte went bankrupt over Mugen is unclear, but the mash up style of adding other fighting game content and media and thus copyrighted material makes Mugen commercially uninteresting. Its amateur community continued anyway.

3. Amateurs creating games

A field where amateur games are still produced from the ground up by one or a few people is the Demoscene. As of today just over a 1000 games are downloadable from the Demoscene website pouet.net.[10]
Although some games are re-releases of demo games from the 90's, release dates vary from 2001 to present day. The Demoscene is a scene of computer programming dedicated to displays of code efficiency and showcasing advanced computer graphics often in combination with software synthesized music. In a paper on putting the Demoscene in perspective Ville-Matias Heikkila says that the Demoscene pioneered in many aspects of "digital subcultures". In order to showcase state of the art computer graphics the Demoscene is "using elements taken from video games and other creative digital works in one's own creations."[11]

[9] http://web.archive.org/web/20031008224842/mugen.elecbyte.com/docs/mugen.html
[10] http://www.pouet.net
[11] Putting the demoscene in a context(p. 3)

Games from the Demoscene have a more close resemblance to the games of the 1980's. They are simple and effective and the game-play is focused on direct control and getting a certain high-score. Coming back to Sherry Turkle's view on the disappearance of computer programming as a method to understanding the mechanics of the computer, the Demoscene is in a way very close to this principle. Upon reviewing the interactivity of works from the Demoscene games Heikkila has the following view:

> Of course, there are also video games released in the demoscene context, but they rarely commit deep excursions into the essence of interactivity. Instead, they tend to focus on the technical issues where the existing strengths of existing demoscene methods can be applied. (Heikkila, 2009, p 7)

However Sherry Turkle refers to a disappearance from mainstream culture, and mainstream is something the Demoscene actively tries not to be. In judging what a good demo is, people within the demoscene are often referring to other demos from the Demoscene itself. This also applies to games. Heikkila gives a good example:

> The video game .kkrieger is a prime example of such an "inter-subcultural" intervention: instead of creating just another size-limited demo, the demoscene group .theprodukkt decided to make a size-limited video game instead. By simply choosing another format that is more accessible to video gaming audience, .theprodukkt managed to critique aspects of the video game culture from inside the video game culture itself. The mere use of demoscene techniques and a tight size limitation served as a statement. (Heikkila, 2009, p 9)

The game ".kkrieger" was far from amateuristic in a technical sense. To have a fully functional game only 96 kilobyte in size was a feat not many professional game development teams could have done. The demoscene group .theprodukkt got some mainstream media attention for .kkrieger and their regular demo releases. Still I feel .kkrieger was very much an amateur release. It was made with love for coding and was released for free. More importantly .kkrieger was critical of modern day programming in video game development. Code was not efficient, programs took up to much space. In a sense this is an anti-commercial stance, most professional game development companies are funded by hardware companies. These hardware companies have little interest in games that are super efficiently coded and thus need less expensive hardware for what they offer.

Like the Demoscene, the Console Homebrew scene also came out of a hacker perspective. Homebrew is a popular term used for applications that are created and executed on a video game console by hackers and consumers. [12]

[12] http://www.wiibrew.org

The Homebrew scene is much more focused on mainstream gaming culture. Most of the games released for a console that has been cracked to run third party or homebrew software are emulated versions of games released on other mostly older platforms but there are amateur made games.
At the moment of writing one of the biggest Nintendo DS homebrew sites, NDSHB.com has 8393 registered users, 64 downloadable games and 84 downloadable games in the outdated section[13]
And this is only one of the homebrew platforms. In my opinion hacking a console to run third party software and creating amateur content for consoles has made a slight comeback to mainstream culture. A short tutorial on how-to run homebrew software on the Nintendo Wii has even been featured on the American cable television channel G4.

4. Conclusion

Although in my research I have seen a lot of different styles and approaches of amateurs that produce games, I have tried to make a selection that makes my definition of amateur game programming more clear. Even though for example the Homebrew scene and Demoscene share their games via the web, I feel that the culture behind game modification is more socially shared. The success of a game mod depends more upon its users enjoying it.

However in a certain way Sherry Turkle is still right in saying that computer programming as a socially shared activity has disappeared from our mainstream culture. The Demoscene and other forms of programming are done by the few. Modified and homebrew games are plentiful in our mainstream culture. However one is left to wonder how many of the users of these games actually program them? Olli Sotamaa said this in his paper on User-centered design practices:

> Exploring games helped to understand the potential of the machines. Tinkering with computers and improving the existing algorithms went hand in hand with playing games. (Olli Sotamaa 2005, p5)

This idea is indeed less represented in the way I see modern day amateur game development. The games of the 1980's were built upon simple code and algorithms, the games of the 1990's and onward were build on graphics, sounds, 3D models and sequences of events.
Perhaps the potential of computers is so omnipresent in our society that the need for understanding this potential has disappeared? Above all else stand the needs and desires of the amateurs themselves. The need to know how a program works has become less vital for the average computer user. A personal way to consume and filter media is something, that seems more urgent today. For personal gain, amateurs are more than willing to go to considerable lengths trying to make a game that suits their own needs or desires.

[13] http://hive.ndshb.com

As the anonymous “Gooseman” a developer of Counter-Strike, a mod for the first person shooter Half life explains[14]

> My initial motivation [for making mods] was probably the same as anyone else involved in the mod scene. I just wanted to customize the game to fit my vision of what a game should be. First and foremost, it is MY vision. Not anyone else's. I don't spend 10+ hours a week working on a mod for free just to make a mod that satisfies everyone, I make a mod that I am happy with and if someone else happens to like it, then that's a bonus(Gooseman, 2001)

When reading this quote to me it seems that in the field of Amateur game programming, software programming as a socially shared activity has not disappeared but the reasons to do it are very different from what they were 25 years ago.

[14] WADs, Bots and Mods: Multiplayer FPS Games as Co-creative Media (p. 6)

References:

Turkle, Sherry(1984 and 2004) “The Second Self” : The Mit Press

Morris, Sue(2004) “WADs, Bots and Mods: Multiplayer FPS Games as Co-creative Media” : Department of English, Media Studies and Art History, University of Queensland

Sotamaa, Olli, "Creative User-centred Design Practices: Lessons from Game Cultures" in Haddon et al. (eds.) Everyday Innovators: Researching The Role of Users in Shapring ICTs. Springer Verlag, London, 2005,page 104-116.

Heikkila, Ville-Matias(2009) “Putting the demoscene in a context”

http://www.gamesetwatch.com

http://web.archive.org

http://en.wikipedia.org/

http://www.wiibrew.org

http://hive.ndshb.com/